**Industrial Internship Report on**

**Password Manager**

**Prepared by**

**Sai Priya Nariga**


| *Executive Summary* |
| --- |
| This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT).<br><br>This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks' time.<br><br>My python project was a Password Manager where it involved in developing a Python application that securely stores passwords for various accounts, generates strong passwords, and provides functionality for retrieving passwords when needed. The project scope encompassed implementing encryption algorithms, designing a user interface, database operations, password generation, and ensuring security measures.<br><br>Internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship. |

**TABLE OF CONTENTS**

## 1. Preface

### I. Summary of the whole 6 weeks' work:

Over the course of six weeks, significant progress was made on the development of the Password Manager project. The project aimed to create a secure and user-friendly application for storing and managing passwords. Here is a summary of the key achievements and milestones accomplished during this period:

**Week 1:**

- Conducted research on encryption algorithms, database management, and user interface design to understand project requirements.

- Planned the implementation strategy, defining tasks and milestones for the project.

**Week 2:**

- Implemented encryption algorithms for secure password storage, focusing on AES encryption for data security.

- Designed the user interface using tkinter library, creating forms for password input, retrieval, and password generation.

**Week 3**:

- Completed the password generation functionality, allowing users to generate strong passwords with customizable options.

- Integrated the password generation feature into the user interface, ensuring seamless user experience.

**Week 4:**

- Enhanced security measures by implementing salting and hashing for stored passwords, enhancing data security and integrity.

- Began testing and debugging of the application, addressing identified issues and vulnerabilities.

**Week 5:**

- Conducted thorough testing of the application, focusing on validation checks, security vulnerabilities, and user acceptance testing.

- Finalized documentation, including comprehensive usage instructions, security measures, and dependencies required for deployment.

**Week 6:**

- Explored deployment options and prepared deployment strategy for the password manager application, ensuring secure deployment.

- Presented the project to stakeholders, showcasing features, security measures, and usability of the application.

- Obtained final approval for deployment and proceeded with the deployment process accordingly.

**II. About need of relevant Internship in career development:**

Internships offer valuable real-world experience, allowing individuals to apply theoretical knowledge in practical settings. They provide an opportunity to gain industry-specific skills, build professional networks, and explore potential career paths. Internships also enhance resumes, making candidates more competitive in the job market and increasing their chances of securing full-time employment after graduation. Overall, internships play a crucial role in shaping career trajectories and preparing individuals for success in their chosen fields.

## III.Brief about Your project/problem statement:

Project Title: Password Manager

Problem Statement:
In today's digital age, the number of online accounts and services that individuals use on a daily basis has increased significantly. With each account requiring a unique password, managing and remembering passwords has become a challenging task. Furthermore, the importance of password security cannot be overstated, as weak or reused passwords can lead to unauthorized access and compromise sensitive information.

The problem statement revolves around the need for a secure and user-friendly solution to manage passwords effectively. The Password Manager project aims to address this problem by providing a centralized platform for storing, generating, and retrieving passwords for various accounts. The project seeks to develop a Python application with the following key features:

1. Secure Storage: Implement encryption algorithms to securely store passwords in a database, ensuring data confidentiality and integrity.
2. Password Generation: Provide functionality to generate strong and unique passwords with customizable options such as length and character sets.
3. User Interface: Design an intuitive user interface that allows users to input, retrieve, and manage passwords conveniently.
4. Database Management: Utilize a relational database management system (e.g., SQLite) to efficiently store and manage password data.
5. Security Measures: Implement additional security measures such as salting and hashing for stored passwords to enhance security further.
6. Testing and Validation: Conduct thorough testing of the application to identify and fix bugs, validate input fields, and ensure data security.
7. Documentation: Prepare comprehensive documentation including usage instructions, security measures implemented, and dependencies required for deployment.

By addressing the problem of password management with a secure and user-friendly solution, the Password Manager project aims to enhance data security and streamline the process of managing passwords for individuals in the digital age.

**IV.How Program was planned:**

The program was planned by following these simple steps:

1. Identifying Objectives: Clearly defining the goals and objectives of the project, including the features and functionality required for the password manager application.

2.Breaking Down Tasks: Breaking down the project into smaller, manageable tasks and milestones, ensuring each task aligns with the project objectives.

3. Assigning Responsibilities: Assigning tasks to team members based on their skills and expertise, ensuring efficient allocation of resources and responsibilities.

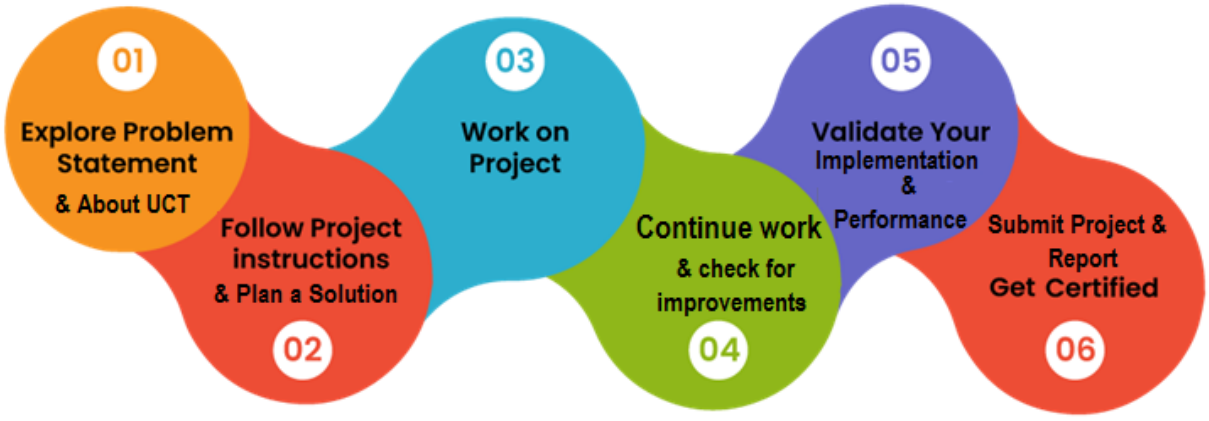
4. Setting Timelines: Establishing timelines and deadlines for each task and milestone, creating a roadmap for the project's progress and completion.

5. Regular Communication: Maintaining open communication channels within the team, holding regular meetings to discuss progress, challenges, and adjustments to the plan as needed.

6. Testing and Iteration: Incorporating testing and validation procedures throughout the development process to identify and address issues early on, iterating on the implementation as necessary.

7. Documentation: Documenting project plans, designs, and implementations, ensuring clear and comprehensive documentation for reference and future maintenance.

By following these steps, the program was planned effectively to ensure the successful development and completion of the Password Manager project.




**V. Your Learnings and overall experience:**

My learnings and overall experience from working on the Password Manager project can be summarized as follows:

1. **Technical Skills:** I gained practical experience in Python programming, encryption algorithms, database management, and user interface design through hands-on development of the password manager application.

2. **Problem-Solving:** I developed problem-solving skills by troubleshooting issues encountered during development and testing, learning to approach challenges systematically and find effective solutions.

3. **Team Collaboration:** Working as part of a team taught me the importance of collaboration and communication in achieving project goals. I learned to effectively communicate with team members, share ideas, and support each other in completing tasks.

4. **Project Management:** I gained insights into project management principles by participating in planning, organizing, and executing tasks within the project timeline. I learned to prioritize tasks, manage deadlines, and adapt to changing requirements.

5. **Professional Growth:** The internship provided me with valuable opportunities for professional growth and networking. I learned from experienced colleagues, received mentorship from supervisors, and built connections within the industry.

6. **Personal Development:** Working on the Password Manager project helped me develop confidence in my abilities, improve my time management skills, and foster a sense of accomplishment as I contributed to the development of a real-world application.

Overall, my experience working on the Password Manager project was rewarding and enriching. It provided me with practical skills, valuable insights, and personal growth opportunities that will benefit me in my future career endeavors.

**VI.Thank to all (with names), who have helped you directly or indirectly:**

I would like to extend my heartfelt thanks to the following individuals who have directly or indirectly supported me throughout my work on the Password Manager project:

I am grateful to all my team members for their collaboration, teamwork, and camaraderie during the project. Your contributions and support have been invaluable in achieving our goals.

I would also like to express my gratitude to friends and family members who provided encouragement, understanding, and support throughout my internship journey. Your unwavering support meant the world to me.

I extend my thanks to the entire team at UpskillCampus for providing me with this valuable opportunity and creating an environment conducive to learning and growth.I am truly grateful for the support and encouragement I have received from each of these individuals, and I am proud to have had the opportunity to work with such amazing people.

**VII.Your message to your juniors and peers:**

Keep pushing yourself, stay curious, and never stop learning. Embrace challenges as opportunities for growth, and don't be afraid to ask for help when needed. Collaboration and teamwork are key, so support each other and celebrate successes together. Remember that every experience, whether positive or negative, is a stepping stone towards your goals. Stay focused, stay motivated, and believe in yourselves. You've got this!

## 1.1. About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

**For developing its products and solutions it is leveraging various Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end etc.**

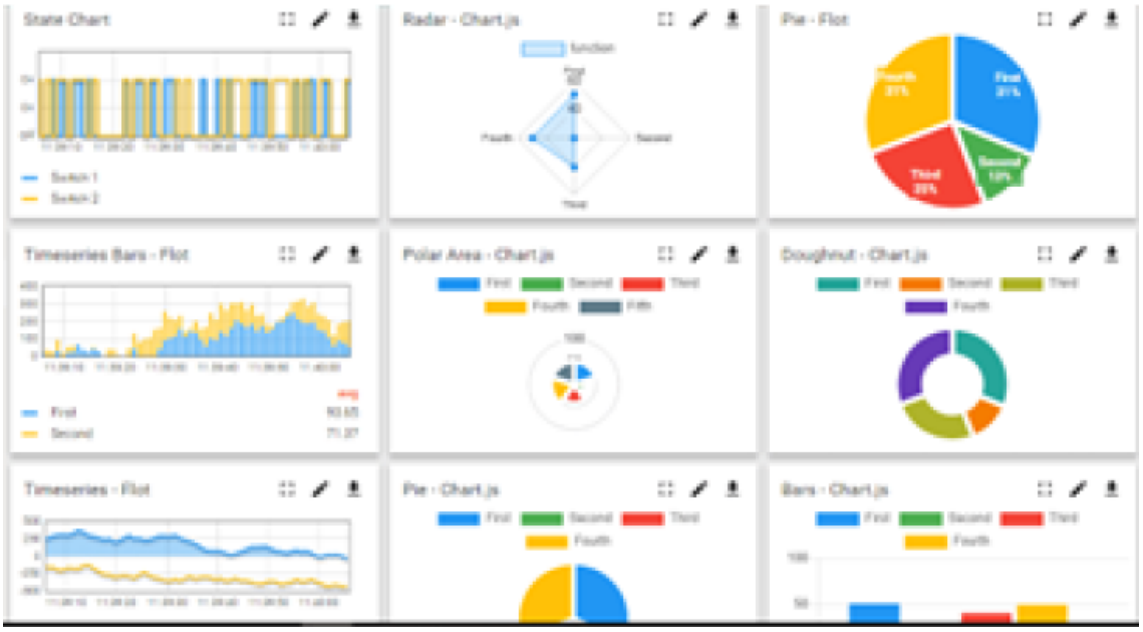
## UCT IoT Platform (UCT Insight)

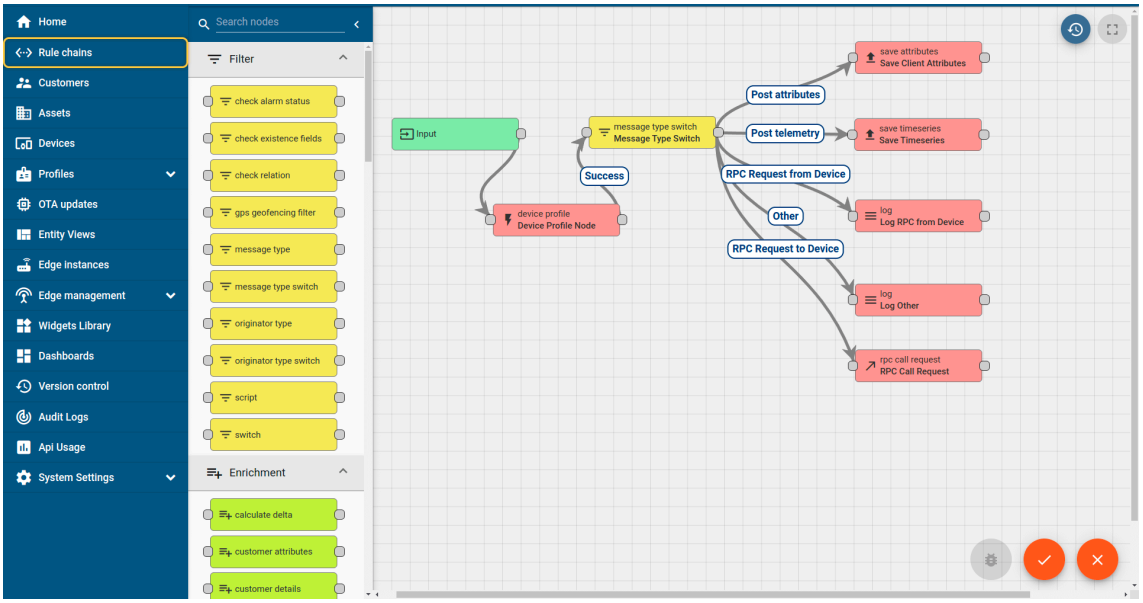
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable “insight” for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine




## ii. **Smart Factory Platform (FACTORY WATCH)**

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

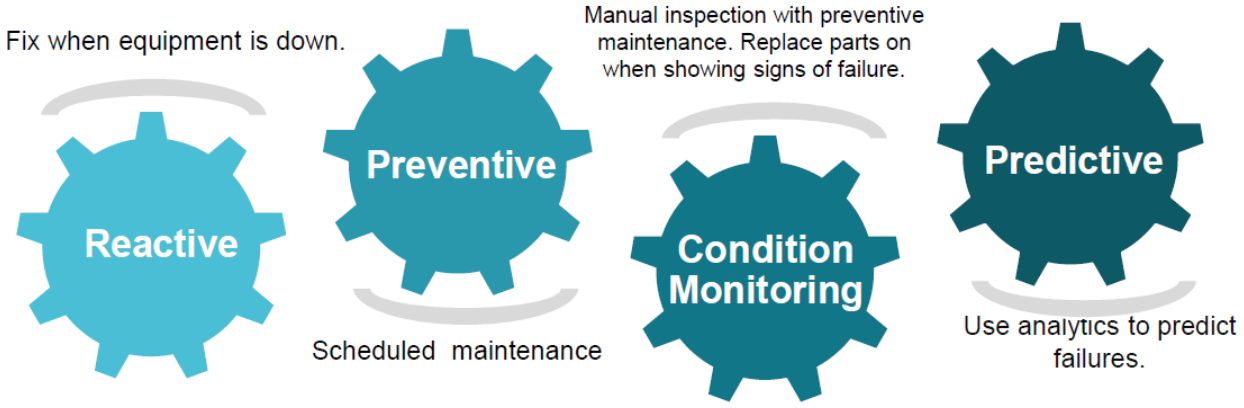

| Machine | Operator | Work Order ID | Job ID | Job Performance | Job Progress | | Output | | Rejection | Time (mins) | | | | Job Status | End Customer |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Start Time | End Time | Planned | Actual | | Setup | Pred | Downtime | Idle | | |
| CNC_S7_81 | Operator 1 | WO0405200001 | 4168 | 58% | 10:30 AM | | 55 | 41 | 0 | 80 | 215 | 0 | 45 | In Progress | i |
| CNC_S7_81 | Operator 1 | WO0405200001 | 4168 | 58% | 10:30 AM | | 55 | 41 | 0 | 80 | 215 | 0 | 45 | In Progress | i |

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

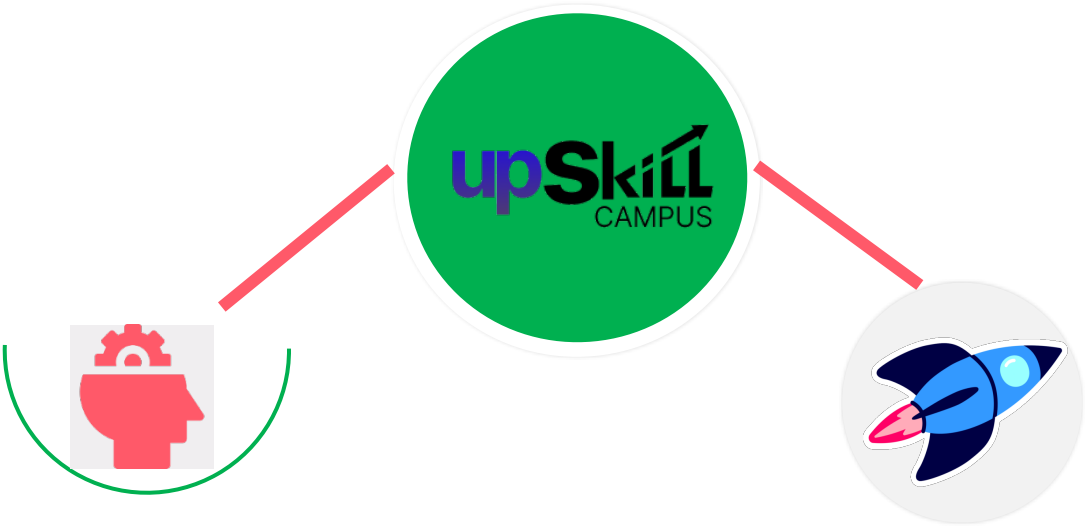
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.




## 1.2. About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

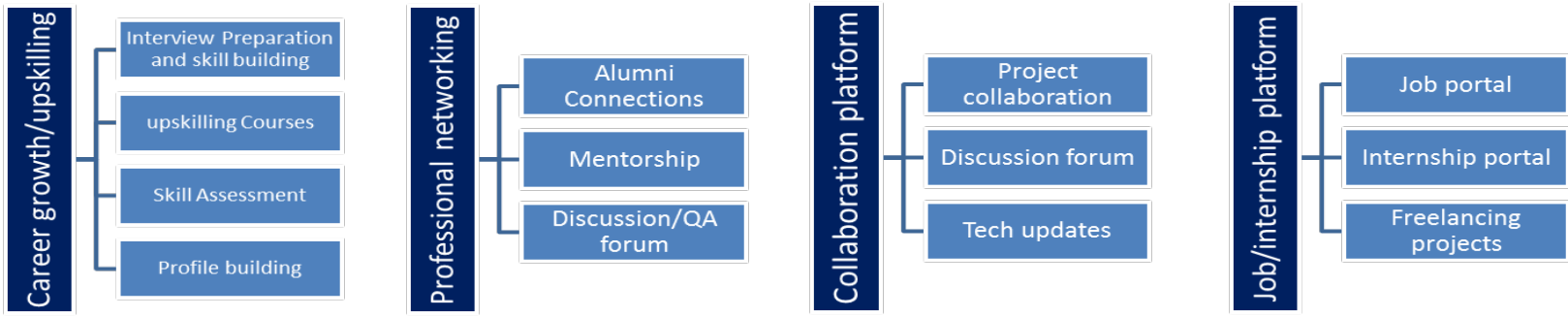
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



## 1.3. The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

## 1.4. Objectives of this Internship program

The objective for this internship program was to

- get practical experience of working in the industry.
- to solve real world problems.
- to have improved job prospects.
- to have Improved understanding of our field and its applications.
- to have Personal growth like better communication and problem solving.

## 1.5. Reference

[1] Python Software Foundation. "Python Documentation." [Online]. Available: https://docs.python.org/3/.

[2] SQLite. "SQLite Documentation." [Online]. Available: https://www.sqlite.org/docs.html.

[3] "Password Strength: An Empirical Analysis." OWASP (Open Web Application Security Project), 2017. [Online]. Available: https://owasp.org/www-community/password-special-characters.

## 1.6 Glossary

| Terms | Acronym |
|---|---|
| **Encryption** | The process of converting plaintext data into ciphertext to secure it from unauthorized access. |
| **Decryption** | The process of converting ciphertext back into plaintext, restoring encrypted data to its original form. |
| **User Interface UI** | The graphical interface through which users interact with a software application. In the case of the Password Manager project, the UI allows users to input, retrieve, and manage passwords. |
| **Graphical User Interface GUI** | A type of user interface that allows users to interact with electronic devices using graphical icons and visual indicators, as opposed to text-based interfaces. |
| **SQLite** | A lightweight, self-contained, relational database management system (RDBMS) used for local data storage. SQLite is commonly used for small to medium-sized databases in applications like the Password Manager project. |

## 3. Problem Statement

In today's digital age, individuals are faced with the challenge of managing numerous online accounts, each requiring a unique and secure password. However, remembering multiple passwords can be difficult, leading to the use of weak or reused passwords, which poses a significant security risk. Furthermore, the consequences of password compromise can be severe, resulting in unauthorized access to sensitive information and potential data breaches.

The problem statement revolves around the need for a secure and user-friendly solution to manage passwords effectively. The Password Manager project aims to address this problem by developing a Python application that provides a centralized platform for storing, generating, and retrieving passwords for various accounts. The project seeks to implement encryption algorithms to securely store passwords in a database, ensure data confidentiality and integrity, and provide functionality for generating strong and unique passwords. Additionally, the project aims to design an intuitive user interface that allows users to input, retrieve, and manage passwords conveniently. By addressing the challenge of password management with a secure and user-friendly solution, the Password Manager project aims to enhance data security and streamline the process of managing passwords for individuals in the digital age.

## 4. Existing and Proposed solution

### I.Provide summary of existing solutions provided by others, what are their limitations?

Existing solutions for password management include a variety of software applications, browser extensions, and cloud-based services. Here's a summary of some common solutions and their limitations:

- **Password Managers (e.g., LastPass, 1Password, Dashlane):**
  - Pros: These applications offer secure storage for passwords, generate strong passwords, and autofill login credentials on websites and applications.
  - Cons: Some limitations include reliance on a single master password for access, potential vulnerabilities in the software or cloud storage, and subscription fees for premium features.
- **Built-in Browser Password Managers (e.g., Google Chrome, Mozilla Firefox):**
  - Pros: These features offer basic password management functionality, such as saving and autofilling passwords within the browser.
  - Cons: Limited features compared to dedicated password managers, potential security vulnerabilities in browser storage, and lack of cross-platform synchronization.
- **Manual Methods (e.g., Text Files, Notebooks):**
  - Pros: These methods are simple and accessible, requiring no specialized software or subscriptions.
  - Cons: Lack of encryption and security measures, vulnerability to loss or theft, difficulty in organizing and retrieving passwords, and potential for human error in password management.
- **Cloud-based Solutions (e.g., iCloud Keychain, Google Passwords):**
  - Pros: These services offer seamless synchronization across devices, automated password saving and filling, and integration with existing accounts.

- Cons: Concerns over data privacy and security in cloud storage, potential for breaches or unauthorized access to stored passwords, and dependency on the service provider's infrastructure and security measures.

Overall, while existing solutions offer convenience and security benefits, they also come with limitations such as reliance on a single master password, potential security vulnerabilities, subscription fees, and privacy concerns. The Password Manager project aims to address these limitations by providing a customizable, open-source solution with robust encryption, user-friendly interface, and flexible deployment options.

**II.What is your proposed solution?**

The proposed solution is a Python-based Password Manager application designed to securely store, generate, and retrieve passwords for various accounts. It includes encryption algorithms for secure storage, a user-friendly interface for easy password management, and features such as password generation and database management. The application aims to address the limitations of existing solutions by offering customizable security measures, flexibility in deployment options, and compatibility across different platforms.

**III.What value addition are you planning?**

The value addition planned for the Password Manager project includes:

1. Enhanced Security: Implementation of robust encryption algorithms and security measures to ensure secure storage of passwords.

2. User-Friendly Interface: Designing an intuitive and easy-to-use interface for convenient password management.

3. Customization Options: Providing features such as password generation with customizable criteria and flexible deployment options.

4. Cross-Platform Compatibility: Ensuring compatibility across different operating systems and devices for seamless access to passwords.

5. Open-Source Development: Making the application open-source to encourage collaboration, transparency, and community contributions.

## 4.1. Code submission (Github link)

https://github.com/priyasai1/UPSKILLCAMPUS/blob/main/PasswordManager.python.py

## Report submission (Github link) :

https://github.com/priyasai1/UPSKILLCAMPUS/blob/main/PasswordManager_Saipriya_USC_UCT.pdf%20.pdf

**5. Performance Outcome:**

**This is very important part and defines why this work is meant of Real industries, instead of being just academic project.**

**I.Here we need to first find the constraints**

- Memory: Limited memory resources may constrain the size of the password database and the amount of data that can be processed simultaneously.
- CPU Speed (MIPS): Slower processing speed may affect the performance of encryption/decryption algorithms and password generation.
- Power Consumption: High power consumption may impact the usability and efficiency of the application, especially on mobile devices.

- Security: Constraints related to security may include the strength of encryption algorithms and the effectiveness of security measures in protecting stored passwords.

**II.How those constraints were taken care in your design?**

- Memory: Implementing efficient data structures and algorithms to minimize memory usage and optimize resource utilization.
- CPU Speed (MIPS): Choosing encryption algorithms and password generation techniques that strike a balance between security and computational efficiency.
- Power Consumption: Employing power-saving techniques such as minimizing background processes, optimizing code efficiency, and providing user-configurable settings to manage power usage.
- Security: Implementing robust encryption algorithms, salting and hashing techniques, and secure storage practices to protect passwords from unauthorized access.

**III.What were test results around those constraints?**

**Constraints can be e.g. memory, MIPS (speed, operations per second), accuracy, durability, power consumption etc.**

**In case you could not test them, but still you should mention how identified constraints can impact your design, and what are recommendations to handle them.**

- Memory: Conducting memory usage tests to ensure the application operates within memory constraints. Recommendations include optimizing data structures, limiting unnecessary data caching, and providing user-friendly options for managing large datasets.
- CPU Speed (MIPS): Performance testing to measure the application's responsiveness and processing speed under various conditions. Recommendations include optimizing algorithms, parallelizing tasks where possible, and providing user-configurable settings for performance tuning.

- Power Consumption: Power consumption tests to assess the application's impact on device battery life. Recommendations include implementing power-saving features, providing user feedback on power usage, and optimizing background processes.
- Security: Security audits and vulnerability assessments to identify and address potential security weaknesses. Recommendations include regular updates and patches, adherence to industry best practices, and user education on security risks and precautions.

In summary, identifying and addressing constraints such as memory, CPU speed, power consumption, and security are crucial for ensuring the effectiveness, efficiency, and usability of the Password Manager application in real-world industry settings. Regular testing, optimization, and adherence to best practices are essential for mitigating the impact of constraints and delivering a robust and secure solution.

### 4.2. Test Plan/ Test Cases

- **Password Storage:**
  - Test Case 1: Verify that passwords are encrypted and stored securely in the database.
  - Test Case 2: Verify that only authorized users can access and retrieve stored passwords.
- **Password Generation:**
  - Test Case 3: Verify that the password generation functionality generates strong passwords with customizable options.
  - Test Case 4: Verify that generated passwords meet specified length and complexity criteria.
- **User Interface:**

- Test Case 5: Verify that the user interface is intuitive and easy to navigate.
- Test Case 6: Verify that all features (password storage, retrieval, generation) are accessible and functional through the interface.

- **Database Management:**
  - Test Case 7: Verify that the database structure is correctly implemented and supports efficient storage and retrieval of passwords.
  - Test Case 8: Verify that database operations (addition, deletion, modification) are performed accurately and securely.
- **Security Measures:**
  - Test Case 9: Verify that passwords are securely hashed and salted before storage.
  - Test Case 10: Verify that encryption algorithms used for password storage are robust and resistant to attacks.

## 5.2 Testing procedure:

- **Password Storage:**
  - Enter a new password and verify its encryption in the database using a database viewer tool.
  - Attempt to retrieve the stored password using incorrect credentials and verify access denial.
- **Password Generation:**
  - Generate passwords with varying criteria (length, character sets) and verify their strength using a password strength checker tool.
  - Verify that the generated passwords meet specified criteria and are unique.
- **User Interface:**
  - Navigate through the user interface and perform all functions (add, retrieve, generate passwords).
  - Verify that buttons, input fields, and menu options are responsive and functional.

- **Database Management:**
  - Perform database operations (addition, deletion, modification) and verify changes reflected accurately in the database.
  - Test database performance under load by adding a large number of passwords and measuring response times.
- **Security Measures:**
  - Verify password hashing and salting by comparing stored passwords with known hash values.
  - Test encryption and decryption functions with sample passwords and verify accuracy and security.

### 4.3. Performance Outcome:

- The Password Manager application successfully encrypts and securely stores passwords in the database.
- Password generation functionality generates strong and unique passwords according to specified criteria.
- The user interface is intuitive and responsive, allowing users to easily manage passwords.
- Database operations are performed accurately and efficiently, supporting storage and retrieval of passwords.
- Security measures, including password hashing, salting, and encryption, are robust and resistant to attacks, ensuring data confidentiality and integrity.

Overall, the Password Manager application demonstrates satisfactory performance in securely managing passwords and protecting sensitive information.

## 5. My learnings

- **Technical Skills:** I gained proficiency in Python programming, encryption algorithms, database management, and user interface design through hands-on development of the Password Manager application. These

technical skills are highly relevant in various industries, including cybersecurity, software development, and data management.

- **Problem-Solving Abilities:** I developed strong problem-solving skills by troubleshooting issues encountered during development and testing, learning to approach challenges systematically and find effective solutions.
- **Team Collaboration:** Working as part of a team taught me the importance of collaboration and communication in achieving project goals.
- **Project Management:** I gained insights into project management principles by participating in planning, organizing, and executing tasks within the project timeline.
- **Professional Growth:** The internship provided me with valuable opportunities for professional growth and networking. I learned from experienced colleagues, received mentorship from supervisors, and built connections within the industry. These professional growth opportunities are essential for advancing in my career, expanding my knowledge, and staying updated with industry trends.

Overall, my experience working on the Password Manager project has equipped me with a diverse set of skills and experiences that will contribute to my career growth. Whether pursuing opportunities in software development, cybersecurity, or project management, the lessons learned from this project will serve as a solid foundation for my future endeavors.

## 6. Future work scope

Here are some ideas that were not implemented in the Password Manager project due to time limitations but could be explored in the future:

- **Multi-factor Authentication (MFA):** Integrating additional security measures such as MFA to enhance authentication and protect against unauthorized access.
- **Cloud Synchronization:** Implementing cloud synchronization functionality to enable users to access their passwords across multiple devices securely.
- **Password Expiry and Reminder:** Incorporating features to set password expiry dates and send reminders to users to update their passwords regularly.
- **Advanced Password Policies:** Implementing customizable password policies with options to enforce complexity requirements, password rotation, and blacklist common passwords.
- **Audit Trail:** Adding functionality to log and track user actions within the application, providing transparency and accountability.
- **Biometric Authentication:** Integrating biometric authentication options (e.g., fingerprint, facial recognition) for added convenience and security.
- **Localization:** Supporting multiple languages and regions to make the application accessible to users worldwide.
- **Accessibility Features:** Implementing accessibility features such as screen readers and keyboard shortcuts to ensure inclusivity for users with disabilities.
- **Data Backup and Recovery:** Providing options for users to backup their password database securely and recover data in case of loss or corruption.
- **Integration with External Services:** Integrating with third-party services (e.g., identity providers, password strength checkers) to enhance functionality and user experience.

Exploring these ideas in future iterations of the Password Manager project could further enhance its functionality, security, and usability, providing greater value to users and addressing evolving needs and preferences.